



# C++

## Programming Step-by-Step



Asadullah Shah



IIUM PRESS

INTERNATIONAL ISLAMIC UNIVERSITY MALAYSIA

# **C++ PROGRAMMING: STEP BY STEP**

---

**Editors**

Asadullah Shah



**IIUM Press**

Published by:  
IIUM Press  
International Islamic University Malaysia

First Edition, 2011  
©IIUM Press, IIUM

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without any prior written permission of the publisher.

Perpustakaan Negara Malaysia

Cataloguing-in-Publication Data

Bibliography p.  
Includes Index  
ISBN

ISBN: 978-967-418-090-4

Member of Majlis Penerbitan Ilmiah Malaysia -- MAPIM  
(Malaysian Scholarly Publishing Council)

Printed by :  
**IIUM PRINTING SDN. BHD.**  
No. 1, Jalan Industri Batu Caves 1/3  
**Taman Perindustrian Batu Caves**  
Batu Caves Centre Point  
68100 Batu Caves  
Selangor Darul Ehsan

# CONTENTS

---

<b>DEDICATION</b>	iii
<b>PREFACE</b>	viii
<b>ACKNOWLEDGEMENT</b>	ix
<b>1. INTRODUCTION</b>	
<i>Asadullah Shah and Assadullah Shaikh .....</i>	<i>1</i>
<b>2. ARITHMETIC EXPRESSIONS AND DATA TYPES IN C++</b>	
<i>Asadullah Shah and Assadullah Shaikh .....</i>	<i>5</i>
<b>3. SENDING THE OUTPUT TO A PRINT FILE</b>	
<i>Asadullah Shah and Assadullah Shaikh .....</i>	<i>11</i>
<b>4. DECISION MAKING: IF-ELSE STATEMENTS AND RELATIONAL OPERATORS</b>	
<i>Asadullah Shah and Assadullah Shaikh .....</i>	<i>17</i>
<b>5. LOGICAL OPERATORS AND SWITCH STATEMENTS</b>	
<i>Asadullah Shah and Assadullah Shaikh .....</i>	<i>25</i>
<b>6. REVIEW, SUMMARY &amp; BUILDING SKILL</b>	
<i>Asadullah Shah and Khamran Khowaza .....</i>	<i>33</i>
<b>7. ITERATIVE STRUCTURES</b>	
<i>Asadullah Shah and Khamran Khowaza .....</i>	<i>39</i>

## **8. THE FOR LOOP**

*Asadullah Shah and Khamran Khowaza* ..... 49

## **9. THE DO-WHILE LOOP**

*Asadullah Shah and Khamran Khowaza* ..... 55

## **10. REVIEW OF VARIABLES, FORMATTING**

*Asadullah Shah and Khamran Khowaza* ..... 59

## **11. REVIEW OF ITERATIVE STRUCTURES**

*Asadullah Shah and Sumbul Khowaza* ..... 63

## **12. POST-TEST AND NESTED LOOPS**

*Asadullah Shah and Sumbul Khowaza* ..... 73

## **13. FUNCTIONS**

*Asadullah Shah and Sumbul Khowaza* ..... 83

## **14. CALL-BY-VALUE AND REFERENCE**

*Asadullah Shah and Sumbul Khowaza* ..... 91

## **15. MORE ON FUNCTIONS**

*Asadullah Shah and Sumbul Khowaza* ..... 99

## **16. STRUCTURES (STRUCT) AND FILES**

*Asadullah Shah and Muniba Shaikh* ..... 111

## **17. ARRAYS**

*Asadullah Shah and Muniba Shaikh* ..... 119

## **18. EXERCISE OF ARRAY**

*Asadullah Shah and Muniba Shaikh* ..... 127

<b>19. READ DATA FROM A FILE</b>	
<i>Asadullah Shah and Mumiba Shaikh</i> .....	137
<b>20. OBJECT ORIENTED PROGRAMMING</b>	
<i>Asadullah Shah and Mumiba Shaikh</i> .....	143
<b>21. SELECTION SORTING</b>	
<i>Asadullah Shah and Syed Ifthar Ali</i> .....	153
<b>22. BUBBLE SORT ALGORITHM</b>	
<i>Asadullah Shah and Syed Ifthar Ali</i> .....	161
<b>23. REVIEW OF ARRAYS</b>	
<i>Asadullah Shah and Syed Ifthar Ali</i> .....	167
<b>24. LINEAR SEARCHING</b>	
<i>Asadullah Shah and Syed Ifthar Ali</i> .....	179
<b>25. BINARY SEARCH</b>	
<i>Asadullah Shah and Syed Ifthar Ali</i> .....	189
<b>26. VECTOR CLASS</b>	
<i>Asadullah Shah and Ejaz Ahmed</i> .....	199
<b>27. POINTERS</b>	
<i>Asadullah Shah and Ejaz Ahmed</i> .....	203
<b>28. FUNCTION POINTERS</b>	
<i>Asadullah Shah and Ejaz Ahmed</i> .....	213
<b>29. POLYMORPHISM AND VIRTUAL FUNCTIONS</b>	
<i>Asadullah Shah and Ejaz Ahmed</i> .....	219

<b>30. C++ REFERENCES</b>	
<i>Asadullah Shah and Ejaz Ahmed</i> .....	223
<b>31. CONST CORRECTNESS</b>	
<i>Asadullah Shah and Osama Mahfooz</i> .....	229
<b>32. MORE ON CONST KEYWORDS</b>	
<i>Asadullah Shah and Osama Mahfooz</i> .....	235
<b>33. GOTO STATEMENT</b>	
<i>Asadullah Shah and Osama Mahfooz</i> .....	241
<b>34. HANDLING ERRORS IN C++</b>	
<i>Asadullah Shah and Osama Mahfooz</i> .....	249
<b>35. STATIC: THE MULTIPURPOSE KEYWORD</b>	
<i>Asadullah Shah and Osama Mahfooz</i> .....	253

# 17. ARRAYS

---

Asadullah Shah and Muniba Shaikh

Department of Computer Science, Faculty of Information and  
Communication Technology, International Islamic University Malaysia,  
Malaysia

## Abstract

An array is used to store and process a **collection of data of the same type**. Let me emphasize that an array can hold **multiple values of the same type**; it cannot hold values of different types (we saw that a struct can do that). An array may also be called a list. If you want to write a program to find the central tendency measure (**mean**) of a set of test scores you need to have a variable that could hold multiple integers instead of assigning an identifier for each score. An array is the answer to your need.

## 17.1 Declaring an Array

Any program that you might write should be general enough to apply to different situations; the program you are about to write should not only let us calculate the mean for examination of paper one but also all subsequent examinations. It needs to decide about maximum number of examination papers to consider and this becomes array size. Arrays are known as static lists because changing the size of an array once fixed you need to change the code and recompile. Before using an array, it has to be declared. To declare an array of 50 scores declare it like any other integer variable, but put a [50] in the square brackets: **int scores[50]**. This tell the system that an array of size 50 and be holding inter values.

Just because we declared an array for 50 elements, there is no need to enter all 50 scores. We can enter any number of scores up to 50.